CLAIMS

What is claimed is:

1. A system comprising:

a compiler to generate object code from a computer program;

a code optimizer to optimize the object code generated by the compiler, the code optimizer including a first device to formulate regions, a second device to select initial regions, a third device to apply code motion, a fourth device to apply tail duplication, and a fifth device to compute UEU(E,R) and DED(X,R), wherein UEU(E,R) represents a number of upward exposed registers at a main entry E of a region R that are used in the region R and DED(X,R) represents a number of downward exposed registers at a main exit X of the region R that are defined in the region R;

a memory to store the compiler and the code optimizer; and

a central processing unit (CPU) cooperatively connected to the memory to execute the compiler and the code optimizer.

2. The system of claim 1, wherein the second device selects initial regions by selecting sub-control flow graphs as regions such that the region starts execution mostly at the main entry and completes mostly at the main exit.

3. The system of claim 1, wherein the fifth device computes UEU(E,R) and DED(X,R) using local information from the region R.

4. The system of claim 1, wherein the third device applies code motion by moving instructions outside the region R into the region R.

5. The system of claim 4, wherein the third device moves instructions outside of the region R into the main entry E and the main exit X of the region R.

6. The system of claim 5, wherein the third device moves instructions outside of the region R into the main entry E and the main exit X of the region R, and later moves the instructions from the main entry E and the main exit X of the region R to other places inside the region R.

7. The system of claim 1, wherein the fourth device applies tail duplication to separate reusable instructions executed along a side entry after selection of initial regions.

8. The system of claim 1, wherein the fourth device applies tail duplication during application of code motion.

9. A method comprising:

selecting initial regions;

computing UEU(E,R) and DED(X,R), wherein UEU(E,R) represents a number of upward exposed registers at a main entry E of a region R that are used in the region R and DED(X,R) represents a number of downward exposed registers at a main exit X of the region R that are defined in the region R;

applying code motion; and

applying tail duplication.

10. The method of claim 9, wherein the selecting initial regions includes selecting sub-control flow graphs as regions such that the region starts execution mostly at the main entry and completes mostly at the main exit.

11. The method of claim 9, wherein the computing UEU(E,R) and DED(X,R) is performed using local information from the region R.

12. The method of claim 9, wherein the applying code motion includes moving instructions outside the region R into the region R.

13. The method of claim 12, wherein the moving instructions outside the region R into the region R includes moving instructions outside of the region R into the main entry E and the main exit X of the region R.

14. The method of claim 13, wherein the moving instructions outside of the region R into the region R further includes moving instructions from the main entry E and the main exit X of the region R to other places inside the region R.

15. The method of claim 9, further comprises applying tail duplication to separate reusable instructions executed along a side entry after selection of initial regions.

16. The method of claim 1, further comprises applying tail duplication during application of code motion.

17. A machine-readable medium comprising instructions which, when executed by a machine, cause the machine to perform operations comprising:

selecting initial regions;

computing UEU(E,R) and DED(X,R), wherein UEU(E,R) represents a number of upward exposed registers at a main entry E of a region R that are used in the region R and DED(X,R) represents a number of downward exposed registers at a main exit X of the region R that are defined in the region R;

applying code motion; and

applying tail duplication.